

System Automation mit Puppet und Foreman

Toni Schmidbauer

1. April 2014

whoami

- ▶ SysAdmin@s-itsolutions
- ▶ toni@stderr.at
- ▶ <http://github.com/tosmi>
- ▶ stderr@jabber.org

Agenda

- ▶ Kurze Umfrage
- ▶ Was ist Puppet?
- ▶ Was ist Foreman?
- ▶ Puppet@s-iTSolutions
- ▶ Was haben wir geplant?

Umfrage

Was ist Puppet?

```
1  class linuxwochen2014 (
2      $ensure = present
3  )
4
5  {
6
6      user { 'linuxwochen':
7          ensure => $ensure,
8          uid => 4711,
9          gid => 4711,
10         }
11
12     package { 'emacs':
13         ensure => installed
14     } ->
15     package { 'vi':
16         ensure => absent,
17     }
18 }
```

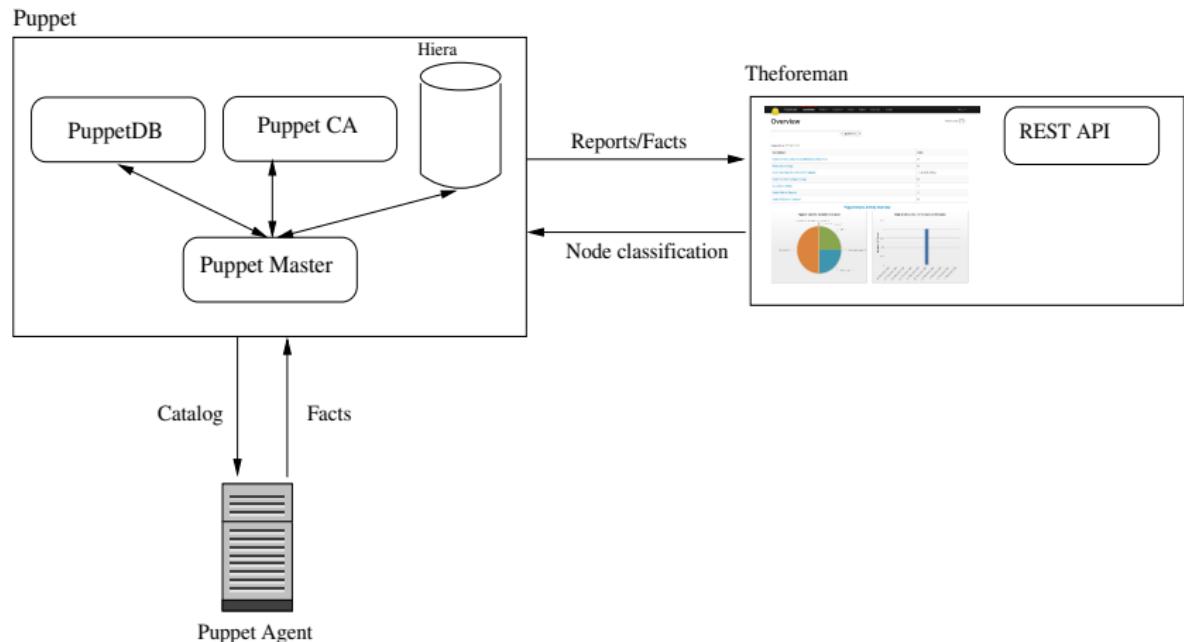
Zuordnung von Klassen

- ▶ über manifests/site.pp

```
1      node linuxwochen {
2          class { 'mypuppetconfig':
3              ensure => absent
4          }
5      }
```

- ▶ über einen External Node Classifier (Foreman)
- ▶ über Hiera (hiera_include('classes',[““]))

Puppet run



Was ist Foreman?

 FOREMAN

Dashboard Hosts Reports Facts Audits Statistics Trends More ▾

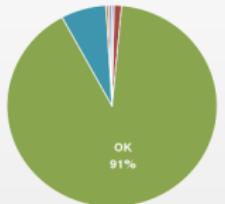
Overview

Filter ... ▾

Host Configuration Status

| | |
|--|-----|
| Hosts that had performed modifications without error | 3 |
| Hosts in error state | 11 |
| Good host reports in the last 35 minutes | 875 |
| Hosts that had pending changes | 0 |
| Out of sync Hosts | 67 |
| Hosts with no reports | 4 |
| Hosts with alerts disabled | 5 |

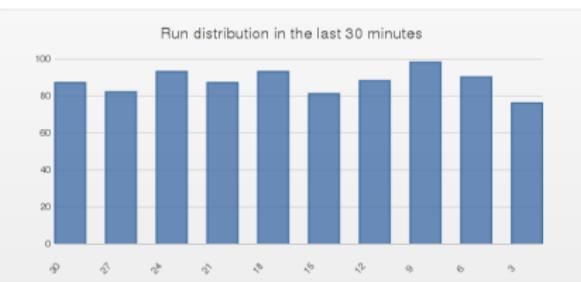
Total Hosts: 965



Latest Events

| Host | A | R | F | FR | S | P |
|----------------|---|---|---|----|---|---|
| appplusu2.l... | 0 | 0 | 0 | 0 | 8 | 0 |
| appqsys2.se... | 0 | 0 | 0 | 0 | 8 | 0 |
| appplusu2.i... | 0 | 0 | 0 | 0 | 8 | 0 |
| appqsys2.se... | 0 | 0 | 0 | 0 | 8 | 0 |
| appplusu2.l... | 0 | 0 | 0 | 0 | 8 | 0 |
| appqsys2.se... | 0 | 0 | 0 | 0 | 8 | 0 |

Run distribution in the last 30 minutes





Und jetzt?



- ▶ Wie verwalten wir unseren Puppet Code?
- ▶ Wie soll unsere Puppet Umgebung aussehen?
- ▶ Wie erfolgt das Deployment des Codes?
- ▶ Wie organisieren wir Module?
- ▶ Wie soll eine Entwicklungsumgebung aussehen?
- ▶ Wie testen wir den Puppet Code?
- ▶ Wie verwalten wir Module von PuppetForge?

Wie verwalten wir unseren Puppet
Code?

Wie organisieren wir Module?

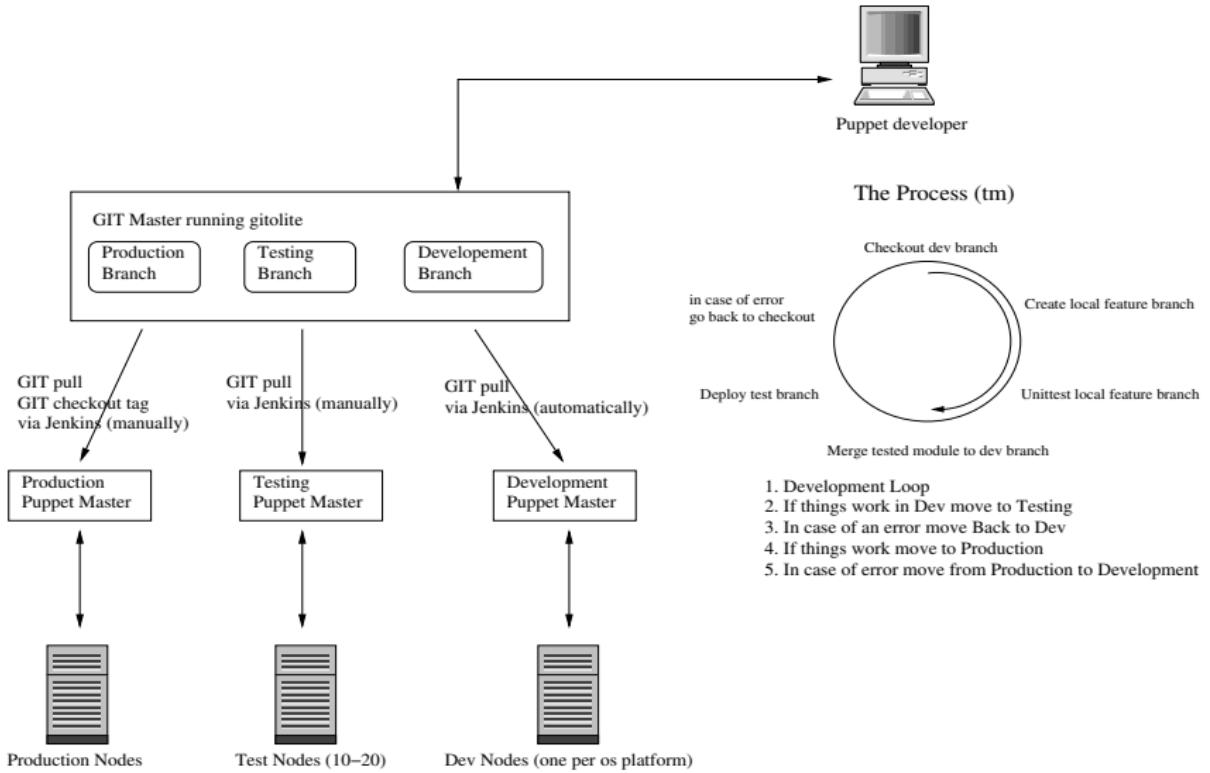
GIT

- ▶ Ein zentrales GIT Repository
- ▶ Berechtigungssystem für GIT mit gitolite
- ▶ 3 Hauptbranches
 - ▶ Master: Staging via GIT pull auf 4 Dev Server
 - ▶ Testing: ca. 25 "Produktions" Server (git pull)
 - ▶ Production: der Rest, Staging via tags
- ▶ Feature Branches für neue Module

Wie soll unsere Puppet Umgebung aussehen?

Wie erfolgt das Deployment des Codes?

Puppet Umgebung



Puppet Umgebung II

- ▶ Am Development Master verwenden wir dynamische Puppet Environments

<http://puppetlabs.com/blog/git-workflow-and-puppet-environments>

- ▶ Jeder Feature Branch wird ein eigenes Puppet Environment
- ▶ Das Staging in Produktion/Testing erfolgt über GIT tags

Deployment

Jenkins | Help us localize this page | Search | Jenkins | Log out | DISABLE AUTO REFRESH

Build Pipeline: Puppet Development

Recent Changes | History | Configure | Add Step | Delete | Merge

| Pipeline | #2760 | #2760 Puppet Development Recipe Tests | #25 Puppet Development List Check | #15 Puppet Development Deployment | #11 Puppet Development Update Foreman Classes |
|----------|-------|---------------------------------------|-----------------------------------|-----------------------------------|---|
| Pipeline | #2759 | #2759 Puppet Development Recipe Tests | #24 Puppet Development List Check | #14 Puppet Development Deployment | #10 Puppet Development Update Foreman Classes |
| Pipeline | #2758 | #2758 Puppet Development Recipe Tests | #23 Puppet Development List Check | #13 Puppet Development Deployment | #9 Puppet Development Update Foreman Classes |
| Pipeline | #2757 | #2757 Puppet Development Recipe Tests | #22 Puppet Development List Check | #12 Puppet Development Deployment | #8 Puppet Development Update Foreman Classes |
| Pipeline | #2756 | #2756 Puppet Development Recipe Tests | Puppet Development List Check | Puppet Development Deployment | Puppet Development Update Foreman Classes |

Page generated: Mar 27, 2014 10:25:12 PM REST API Jenkins ver. 1.532.2

Monitoring

Monitor 

| Puppet Development Deployment | Puppet Development Lint Check |
|---|---|
| #15 0s | #35 9s |
| Puppet Development Rspec Tests | Puppet Development Update Foreman Classes |
| #2760 1m 19s | #11 45s |
| Puppet Production | Puppet Testing Lint Check |
| Possible Culprits: wce, toni, nenioscio #2722 43s | #2 16s |
| Puppet Testing Rspec Tests | |
| #2730 58s | |

Brought to you by 

Wie soll eine Entwicklungsumgebung aussehen?



Vagrant

- ▶ <http://vagrantup.com>
- ▶ Ermöglicht virtuelle Entwicklungsumgebungen
- ▶ Vagrant Box ist ein vorkonfiguriertes Image
- ▶ Default VirtualBox andere Provider via Plugins (VMWare, KVM)

Demo

Wie testen wir den Puppet Code?

rspec-puppet

- ▶ Ruby RSpec Tests für Puppet
- ▶ Ist in unserer Vagrant Umgebung vorinstalliert
- ▶ Jedes Module muss RSpec Tests mitbringen
- ▶ Commit in Master Branch löst einen Jenkins Build aus
- ▶ Commit in Testing Branch löst einen Jenkins Build aus
- ▶ Commit in Production Branch löst einen Jenkins Build aus

Wie verwalten wir Module von PuppetForge?

Puppetforge Module

- ▶ Eigenes GIT Repository (puppetforge.git)
- ▶ Download der Module in der Entwicklungsumgebung
- ▶ Staging wie unser Haupt Puppet Repository
- ▶ Dies ändert sich allerdings (dazu später)

- ▶ Wie verwalten wir unseren Puppet Code? *DONE*
- ▶ Wie soll unsere Puppet Umgebung aussehen? *DONE*
- ▶ Wie erfolgt das Deployment des Codes? *DONE*
- ▶ Wie organisieren wir Module? *DONE*
- ▶ Wie soll eine Entwicklungsumgebung aussehen? *DONE*
- ▶ Wie testen wir den Puppet Code? *DONE*
- ▶ Wie verwalten wir Module von PuppetForge? *DONE*



Probleme, Probleme, Probleme...

- ▶ Ein GIT Repo funktioniert nicht bei Änderungen von Upstream Modulen
- ▶ Andere Abteilungen sollen ihre Module unabhängig testen
- ▶ Unitests sagen noch nichts aus wie sich der Code am Live-System verhält
- ▶ Wir sollten eigentlich das Zusammenspiel aller Module testen (Forge und eigene)

Was haben wir geplant?

- ▶ r10k für Deployment
- ▶ Ein Repository pro Module
- ▶ Nur interne Module bleiben im Hauptrepo
- ▶ Acceptance Tests mit Beaker